UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/675,909	09/30/2003	George Duncan Pearson	CA920085083US2	2080

51871 7590 02/16/2011

Shumaker & Sieffert, P.A.
1625 Radio Drive, Suite 300
Woodbury, MN 55125

EXAMINER	
LAN, TZU-HSIANG	

ART UNIT	PAPER NUMBER
3623	

NOTIFICATION DATE	DELIVERY MODE
02/16/2011	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

pairdocketing@ssiplaw.com

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Application Number: 10/675,909
Filing Date: September 30, 2003
Appellant(s): PEARSON ET AL.

Christopher D. Karlen
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 1/4/2011 appealing from the Office action mailed July 8, 2010.

(1) Real Party in Interest

The examiner has no comment on the statement, or lack of statement, identifying by name the real party in interest in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The following is a list of claims that are rejected and pending in the application:

7-10, 12, 13 and 22-31.

(4) Status of Amendments After Final

The examiner has no comment on the appellant's statement of the status of amendments after final rejection contained in the brief.

(5) Summary of Claimed Subject Matter

The examiner has no comment on the summary of claimed subject matter contained in the brief.

(6) Grounds of Rejection to be Reviewed on Appeal

The examiner has no comment on the appellant's statement of the grounds of rejection to be reviewed on appeal. Every ground of rejection set forth in the Office action from which the appeal is taken (as modified by any advisory actions) is being maintained by the examiner except for the grounds of rejection (if any) listed under the subheading "WITHDRAWN

REJECTIONS." New grounds of rejection (if any) are provided under the subheading "NEW GROUNDS OF REJECTION."

(7) Claims Appendix

The examiner has no comment on the copy of the appealed claims contained in the Appendix to the appellant's brief.

(8) Evidence Relied Upon

20070179828	Elkin et al.	8-2007
6308163	Du et al.	10-2001

Adaytum software, www.adaytum.co.uk 03/40/2001 [retrieved 12/05/05], pages 1-41, retrieved from Archive.org and google.com

J.J. Halliday et al., "Flexible workflow management in the OPENflow system," in 4th International Enterprise Distributed Object Computing Conference (EDOC 2001)

Petra Heinl et al.. "A comprehensive Approach to Flexibility in Workflow Management Systems," SIGSOFT Software Engineering Notes, vol. 24, no. 2, p. 78 (March 1999)

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claim Rejections - 35 USC § 103

1. The following is a quotation of 35 U.S.C. 103 (a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

2. **Claims 7-10, 12-13, and 22-23** are rejected under 35 U.S.C. 103(a) as being unpatentable over Adaytum Software ("Adaytum") in view of Elkin et al. (U.S. 2007/0179828), in view of Halliday and further in view of Heinl.

As for claim 7 Adaytum discloses software for modifying enterprise planning model including:

a enterprise planning program, executable by a computing device, for an enterprise planning session in accordance with an enterprise planning model, wherein, the enterprise planning model defines hierarchically arranged nodes associated with business logic software modules and enterprise contributor (pp. 17-20 especially ¶18 where it discusses that Adaytum Planning module can automatically loads organization hierarchy and live data directly from any general ledge system further pp. 4-5 show that Adaytum software is designed for multiple levels of drill down in a node-level execution environment), wherein executing the enterprise session comprising:

receiving contribution data provided by the enterprise contributors, wherein the contribution data corresponds to one or more of the nodes of the enterprise planning model (Adaytum page 30-32, and 34 i.e. receiving operational data such as billing information from contributor);

reconciling the contribution data across an enterprise that corresponds to the enterprise planning model by automatically aggregating the contribution data (Adaytum, page 30-31 and 34).

While Adaytum discloses all the limitations above, Adaytum fails to explicitly teach the following limitations, however, Elkin discloses:

executing, by a computing device, an enterprise session in accordance with an enterprise model, wherein the enterprise model defines hierarchically arranged nodes associated with business logic software modules and enterprise contributors, wherein executing the enterprise planning session comprises:

receiving contribution data and automating reconciliation of the contribution data corresponds to the enterprise planning model by automatically aggregating the contribution data as the contribution data is received and wherein the enterprise planning model comprises a financial model (see ¶¶ 13, 15, 40-41, 102-106 Fig. 3, Fig. 7-9, noting an enterprise model application allows users to define enterprise models in a hierarchical fashion, for example a mortgage financial model, data aggregation is shown in ¶ 194-198 i.e. deployment package is automatically aggregated and installed as its received [in real-time]);

checking-out an individual model for editing during execution of the enterprise planning session in accordance with the enterprise planning model (see Table 1 - list of operations, including checkout and check in); and

modifying a model without preventing execution of the enterprise planning session for the model (see ¶ 13, noting users may edit the enterprise model without affecting current enterprise operations).

While Elkin and Adaytum teach all the limitations above, they fail to explicitly teach the following limitation, however Heinl teaches:

at least one of the nodes of a workflow model that is not checked out receives contribution data from the checked-out individual one of the nodes without taking the model offline (pp. 85-86 i.e. dirty read function will enable update of an online node with data from an offline node)

Besides limitations disclosed above, examiner previously took official notice that it is old and well known to:

- Modify individual nodes of the model (Heinl, § 2.2, noting that flexibility by adaption is where a node is modified to include additional paths to other nodes; Halliday at 7, noting that tasks are individual nodes of a workflow and can be edited in a dynamic reconfiguration).
- Modify the nodes of the model without preventing execution of the session for the nodes that are not checked out (e.g., execute the enterprise planning session in accordance with the model while the modifications are occurring [i.e. dynamic modification/reconfiguration]). (Heinl at 80, first column, noting that modifications to the workflow occur in real-time, that is, while the model is still running; Halliday § 2.3 - Flexibility by adaptation: Dynamic Reconfiguration, noting tasks are modified).
- Check out individual one of the nodes (e.g., that the administration console allows an analyst to check-out individual nodes of the model for editing during execution of the enterprise planning session without taking the model offline). (Heinl, Fig. 6, noting the lock / check out policy of one node at a time A or B; § 4.2.3, noting the use of check-in / check-out).

It would have been obvious to a person having ordinary skill in the art at the time of invention to modify the enterprise planning system disclosed by Adaytum with Elkin, Halliday, and Heinl's disclosure of enterprise node level operation. Both prior arts acknowledge that reconciliation of business data across enterprise hierarchical nodes is old and well known. Adaytum introduced multiple levels of drill down detail analysis executed by enterprise planning module, but Adaytum fails to explicitly teach detailed node level execution such as node check-out step. Elkin in view of Halliday and Heinl teach detail node level operation between software modules. Since, claimed invention is merely a combination of old elements, and in the combination each element merely would have performed the same function as it did separately i.e. nodal operation mechanism disclosed by Elkin, Heinl and Halliday will perform the same node level operation in the enterprise planning system as in any other nodal workflow environment; hence, one of ordinary skill in the art would have recognized that the results of the combination were predictable.

As for claim 8, see the discussion in claim 7 above. While Adaytum teaches all the limitations above, Adaytum fails to explicitly teach the following, however, Elkin further teaches:

receiving updated model information for a node, and updating a respective slice of the enterprise planning model for only one of the nodes based on the updated model information (see ¶¶ 146 and 196, noting an updated process model may be overlaid on the existing process model in real-time).

Elkin fails to explicitly disclose that the nodes are checked-out. However, as shown in claim 7, node check-out is an old and well-known concept. Therefore, it would have been

obvious to a person having ordinary skill in the art at the time of invention to specify that the updates to the model in Elkin are based on the nodes checked-out for updates, for the purpose of allowing changes to an existing model.

Further, It would have been obvious to one of ordinary skill in the art at the time of the invention to modify Adaytum with Elkin since claimed invention is merely a combination of old elements, and in the combination each element merely would have performed the same function as it did separately i.e. software operation discussed by Elkin, Heinl and Halliday will perform the same function in the enterprise planning system as in project management or mortgage approval system, one of ordinary skill in the art would have recognized that the results of the combination were predictable.

As to claim 9, see the discussion in claim 8 above. While Adaytum teaches all the limitations above, Adaytum fails to explicitly teach the following, however, Elkin further teaches:

wherein updating the enterprise planning model comprises modifying the business logic software module or the enterprise contributor associated with the checked-out individual one of the nodes in response to the updated model information (see ¶¶ 196-198).

Elkin fails to explicitly disclose that the nodes are checked-out. However, as shown in claim 7, node check-out is an old and well-known concept. Therefore, it would have been obvious to a person having ordinary skill in the art at the time of invention to specify that the updates to the model in Elkin are based on the nodes checked-out for updates, for the purpose of allowing changes to an existing model.

Further, It would have been obvious to one of ordinary skill in the art at the time of the invention to modify Adaytum with Elkin since claimed invention is merely a combination of old elements, and in the combination each element merely would have performed the same function as it did separately i.e. software operation discussed by Elkin, Heinl and Halliday will perform the same function in the enterprise planning system as in project management or mortgage approval system, one of ordinary skill in the art would have recognized that the results of the combination were predictable.

As to claim 10, see the discussion in claim 8 above. Adaytum further teaches:

receiving and processing the contribution data from the enterprise contributors associated with the nodes of the model during the execution of the enterprise planning session and prior to the check-out of the individual one of the nodes (Adaytum page 30-31 and 34, i.e. contribution data are received and processed from bottom-up contributors).

While Adaytum teaches all the limitations above, Adaytum fails to explicitly teach the following, however, Elkin further teaches:

updating data of the checked-out model with the contribution data in accordance with the updated model information when the check-out model is subsequently checked-in during the execution of the enterprise planning session (¶ 159-167 and table 1 i.e. checked out model is updated with and edited, then subsequently checked-in, further, ¶179-185 i.e. current edited task is later checked-in while planning session still running).

However, Elkin fails to explicitly disclose that an individual node is checked-out. However, as shown in claim 7, node check-out is an old and well-known concept. Therefore it

would be obvious to one with ordinary skill in the art to check out a node and update the node before check-in the node. Further Elkin does not explicitly teach receiving and processing contribution data from enterprise contributor.

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify Elkin with Adaytum in view of examiner's official notice since claimed invention is merely a combination of old elements, and in the combination each element merely would have performed the same function as it did separately, and one of ordinary skill in the art would have recognized that the results of the combination were predictable.

As to claim 12, see the discussion in claim 10 above. Adaytum further teaches:

defining reconciliation jobs for execution by an application server to prompt a reviewer to reconcile the previously received contribution data with the updated model information for the check-in individual one of the nodes (Adaytum, page 31-32 and 34 i.e. e.planning application able to prompt a reviewer to reconcile the previous data with updated model information for example, managers can input data that update a node and when check-in individual one of the nodes, the system from bottom-up reconciling the contribution data), wherein the application server is communicatively coupled to the computing device (Adaytum page 4-6).

As to claim 13, see the discussion in claim 10 above. Adaytum further teaches:

defining reconciliation jobs for execution by remote computers of the enterprise contributors to prompt at least one of the enterprise contributors to reconcile the previously received contribution data with updated model information for the checked-in individual one of

the nodes (Adaytum page 31-32 and 34 i.e. top-down planning where updated model information is escalated and reconciled).

As to claim 22, see the discussion in claim 7 above. Adaytum further teaches:

receiving a portion of the contribution data (page 31);

identifying higher levels of the hierarchically arranged nodes affected by the portion of the contribution data (Adaytum page 31-32); and

calculating new aggregate totals at each level of the hierarchically arranged nodes according to the received portion (Adaytum page 31-32 i.e. information gap is readily identified for each level of the hierarchically arranged nodes according to either top down or bottom up input).

While Adaytum teaches all the limitations above, Adaytum fails to explicitly teach the following, however, Elkin further teaches:

processing data in real time (¶ 195-198)

It would have been obvious for one of ordinary skill in the art at the time of the invention to modify Elkin with Adaytum in view of examiner's official notice because implementing real-time update would enhance information synchronization across the enterprise.

As to claim 23, see the discussion in claim 12 above. Adaytum further teaches:

receiving an indication from the reviewer corresponding to the checked-in individual one of the nodes (Adaytum page 31 i.e. an administration tool will ensure submission of right information on time, which implies that indication for wrong information would be filtered);

wherein an indication indicates whether the reviewer accepted or rejected the contribution data for the checked-in individual one of the nodes (page 31 i.e. right information is ensured which implicitly express administration tool filters contribution data for the checked-in one of the nodes).

As to claims 24 and 28, Adaytum and Elkin disclose the claimed invention substantially. All the limitation of claim 24 are of the same scope as the limitations of claim 1, and are therefore rejected on the same basis, with following noted exceptions that are further address by Adaytum, Elkin, Heinl and Halliday. Here, Adaytum further teaches:

a computer readable medium (pp. 7-8) comprising:

associating a first set of data with the first node and a second set of data with second node (pp. 11 and 30-31 i.e. node at different level of enterprise has different data set i.e. department level node and managerial level node have different data). While Adaytum implicitly teaches hierarchical node and data relationship (pp. 17-20 especially pp.18); Examiner would like to bring in a teaching reference to show that hierarchical node and data relationship is old and well known (US Patent 6308163 see at least figure 2-3)

While Adaytum teaches the limitation above, Adaytum fail to explicitly teach the following, however, Elkin further teaches:

receiving an update to the enterprise planning model, wherein the update identifies the first node (¶ 162-163);

checking-out, the first node after receiving the update to the enterprise model (¶ 161-166, 168 and table 1 zoom in and check out the first node for edit); and

Heinl further teaches:

checking-in, a modified version of the first node after modifying the second set of data for the second node, wherein the modified version of the first node corresponds to the received update to the enterprise planning model (pp. 84-85 especially under 4.2.2 where first node is checked in and received update to the enterprise model to resolve incompatibility); and

reconcile the modified second set of data with the modified version of the first set of data after first node has been checked in (pp. 85-86 i.e. user of the system will be able to user dirty read function to read locked workflow type although it might create inconsistencies for the online model, the correction can be made when first node checked in and the modified version of first set of data will update modified second set of data as shown in figure 7)

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify Adaytum with Elkin, Heinl, and Halliday because Adaytum introduced multiple levels of drill down detail analysis executed by enterprise planning module, but Adaytum fails to explicitly teach detailed node level execution. However, Elkin, Heinl, and Halliday show that node level modification as disclosed in the claim is old and well known. Since, claimed invention is merely a combination of old elements, and in the combination each element merely would have performed the same function as it did separately i.e. node level update and modification discussed by Elkin, Heinl and Halliday will perform the same function in the enterprise planning system as in other nodal management environment, one of ordinary skill in the art would have recognized that the results of the combination were predictable.

AS to claims 25 and 29, see discussion in claim 24 and 28 above. Adaytum further teaches aggregate data from bottom-up and top-down then reconcile over data to achieve final optimization (pp. 30-31), however, Adaytum fails to explicitly disclose the following.

Elkin further teaches node architecture layout wherein the first node comprises a first childe node, wherein the second node comprises a second child node, wherein the hierarchically arranged nodes further comprise a parent node, wherein the enterprise model defines the parent node as a parent to the first child node and the second child node (fig. 8, 9 and 14-15 shows parent/childe node relationship structure), and

Halliday further teaches:

Aggregating data from the first child node and the second child node to form a set of aggregate data set (pp. 4-8 i.e. workflow shows that child node's data are aggregated into data sets from different childe nodes)

Heinl further teaches:

associating the set of aggregate data with the parent node (pp. 84-85 i.e. update nodes are associated with old workflow nodes)

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify Adaytum with Elkin, Heinl, and Halliday because Adaytum introduced multiple levels of drill down detail analysis executed by enterprise planning module, but Adaytum fails to explicitly teach detailed node level execution. However, Elkin, Heinl, and Halliday show that node level modification as disclosed in the claim is old and well known. Since, claimed invention is merely a combination of old elements, and in the combination each element merely would have performed the same function as it did separately i.e. node level update and

modification discussed by Elkin, Heinl and Halliday will perform the same function in the enterprise planning system as in other nodal management environment, one of ordinary skill in the art would have recognized that the results of the combination were predictable.

As to claims 26 and 30, see the discussion in claim 24 and 28 above. While Adaytum discloses all the limitations above, Adaytum fails to explicitly disclose the following limitation, however, Heinl further discloses:

receiving a second set of contribution data for the first node before checking-out the first node (pp. 85-86 i.e. cooperative model receives a set of data after modification);

defining a reconciliation job that is configured to cause the application server to prompt to a reviewer to reconcile the second set of contribution data with the modified version of the first node (pp. 85-86 i.e. user needs to decide between 3 options including overwrite, save new version, or save variant);

receiving a response from reviewer indicating acceptance or rejection of the second set of contribution data for the modified version of the first node (pp. 85-86 i.e. user's choice will result in different workflow structure that indicates acceptance or rejection of second set of the contribution data)

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify Adaytum with Elkin, Heinl, and Halliday because Adaytum introduced multiple levels of drill down detail analysis executed by enterprise planning module, but Adaytum fails to explicitly teach detailed node level execution. However, Elkin, Heinl, and Halliday show that node level modification as disclosed in the claim is old and well known. Since, claimed

invention is merely a combination of old elements, and in the combination each element merely would have performed the same function as it did separately i.e. node level update and modification discussed by Elkin, Heinl and Halliday will perform the same function in the enterprise planning system as in other nodal management environment, one of ordinary skill in the art would have recognized that the results of the combination were predictable.

As to claims 27 and 31, see the discussion in claim 24 and 28 above. While Adaytum discloses all the limitations above, Adaytum fails to explicitly disclose the following limitation, however, Heinl further discloses:

updating a slice of the workflow model corresponding to the first node to form the modified version of a first child node while the first child node is checked out (Heinl at 80, first column, noting that modifications to the workflow occur in real-time, that is, while the model is still running; Halliday § 2.3 - Flexibility by adaptation: Dynamic Reconfiguration, noting tasks are modified and pp. 85-86 shows modified version of different child node i.e. variant).

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify Adaytum with Elkin, Heinl, and Halliday because Adaytum introduced multiple levels of drill down detail analysis executed by enterprise planning module, but Adaytum fails to explicitly teach detailed node level execution. However, Elkin, Heinl, and Halliday show that node level modification as disclosed in the claim is old and well known. Since, claimed invention is merely a combination of old elements, and in the combination each element merely would have performed the same function as it did separately i.e. node level update and modification discussed by Elkin, Heinl and Halliday will perform the same function in the

enterprise planning system as in other nodal management environment, one of ordinary skill in the art would have recognized that the results of the combination were predictable.

(10) Response to Argument

In response to applicant's argument that the combination of Adaytum-Elkin-Halliday-Heinl fails to disclose "modifying a check-out individual one of nodes of an enterprise planning model without preventing execution of an enterprise planning session for the nodes of the enterprise planning model that are not checked-out, wherein at least one of the nodes of the enterprise planning model that is not checked out receives contribution data from the checked-out individual one of the nodes, without take the model offline", examiner respectfully disagree.

Throughout applicant's argument section (pp.9-13), applicant fails to consider that the test for obviousness is not whether the features of a secondary reference may be bodily incorporated into the structure of the primary reference; nor is it that the claimed invention must be expressly suggested in any one or all of the references. Rather, the test is what the combined teachings of the references would have suggested to those of ordinary skill in the art. See In re Keller, 642 F.2d 413, 208 USPQ 871 (CCPA 1981). By discussing cited prior arts individually, applicant fails to appreciate the combined teaching suggested by the cited prior arts.

Here, Adaytum software shows capability to perform top-down and bottom up data reconciliation and it operates based on different enterprise hierarchical tier. Further, Adaytum is able to receive contribution data corresponds to one or more of the nodes of the enterprise planning model (see at least pp. 30-32 where contributor able to input day to day operational data). However, Adaytum fail to explicitly teach node level operation such as check-in and

check-out nodes for modification. Elkin, Heinl, and Halliday, on the other hand, disclose detailed node level operation and hierarchically arranged nodes associated with business logic software modules and enterprise distributor (Elkin, fig. 7-9 and 14-15 display hierarchically arranged nodes associated with business logic software module such as receiving module, and claim view module that can be linked to different level of enterprise contributor such as management level contributor and operation level contributor).

Examiner respectfully disagrees with applicant's assertion that the combined teaching of the cited references fails to teach "one node of enterprise model that is not checked out receives contribution data from a checked out node of an enterprise model without taking model offline." As discussed above, Adaytum teaches node level infrastructure in an enterprise planning model, but Adaytum fails to teach the detailed node level operation. Heinl cures the deficiency by teaching a node level operation for online node to receive data from a checked-out node (See at least pp. 85 col. 2 ¶1 and pp. 86, col. 2 ¶5, where allowing "dirty reads" one or more of online nodes will be able to read and receive partial data from a checked-out node.) While applicant asserts that the mechanism of checking out workflow discussed in Heinl section 4.2.2. is different from checking out node in an enterprise model, examiner respectfully disagrees. Adaytum and Heinl both operate under a computational environment where the data communication is based on its infrastructure's technical design. It would have been obvious to one of ordinary skill in the art at the time of the invention to design a node as a representation of a workflow type if one so desires to do so. Further implementing data retrieval mechanism disclosed by Hienl with Adaytum's enterprise resource planning model is merely a combination of old elements, and in the combination each element merely would have performed the same

function as it did separately i.e. enable Adaytum software to retrieve data from checked out node, one of ordinary skill in the art would have recognized that the results of the combination were predictable.

While applicant attempts to show incompatibility for modifying an enterprise planning model with a workflow model, applicant's explanation is not persuasive. Specifically, examiner respectfully disagrees with applicant's assertion that "workflow type of Heinl is not associated with a set of data and cannot receive contribution data" (Appeal Brief pp.10). On page 85, section 4.2.2 Heinl explicitly teaches that users may edit data associated with specific workflow type. The combined teaching of Adaytum and Heinl not only present commonality in enterprise planning model and workflow model i.e. same node to node data communication mechanism, but also show that both model can channel data based on the designed structure. Hence, it would have been obvious to one of ordinary skill in the art at the time of the invention to modify enterprise planning model with workflow operation mechanism.

As to applicant's argument regarding interdependency between workflow types, see the discussion for test of obviousness above. Here, Adaytum and Elkin explicitly teach the interdependency between hierarchically nodes (to further show this element as old and well known, a teaching reference was cited by examiner for the office action filed on 7/28/10, pp.5, see fig 2-3 of US 6308163 of Du et al.). It would have been obvious to one of ordinary skill in the art at the time of the invention to modify Adaytum and Elkin with data contribution mechanism disclosed by Hienl because it is within the capability of one of ordinary skill in the art to apply known node level contribution mechanism with the predictable result of contributing relevant data based on interdependency.

Finally, although the issue of "type" of data being contributed from a check-out node had been discussed in the Office Action 7/28/10, response to argument section, examiner would provide further detail to address applicant's concern. In light of the broadest reasonable interpretation consistent with the specification, contribution data can be understood as any data one can obtain that corresponds to one or more nodes (applicant fails to provide narrower definition for the term "contribution data"). Comparing the teaching of Heinl 4.2.2-4.2.3 (pp.85-86), with applicant's claimed invention the difference merely reside in non-descriptive data label. One of ordinary skill in the art may label data obtained from a "dirty read" as contribution data and depository information i.e. data relating workflow type A and B both stored in same database (fig 6) as model data if one so desires to do so. Regardless of how one wants to label the obtainable data (such as data related to insurance claim [Elkin, ¶47]) and the non-obtainable data, the functionality for obtain partial data from a checked out node remains the same (Heinl, pp.85-86 i.e. dirty read mechanism). Since the claimed invention is merely a combination of old elements, and in the combination each element merely would have performed the same function as it did separately i.e. mechanism of obtaining partial data operate the same in any node based operation environment, one of ordinary skill in the art would have recognized that the results of the combination were predictable.

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

/TZU-HSIANG (SEAN) LAN/

Examiner, Art Unit 3623

/Beth V. Boswell/

Supervisory Patent Examiner, Art Unit 3623

Conferees:

Beth V. Boswell /bvb/

Supervisory Patent Examiner, Art Unit 3623

/Vincnt Millin/

Appeals Practice Specialist